Individual Final Report

1. Introduction

In the field of machine learning, facial expression recognition has been well developed due to the improvement of database and deep learning techniques. Compared to traditional laboratory-controlled database such as the Extended CohnKanade (CK+)[1] and the MMI database[1], new web database includes larger amount of expression pictures from real world, thus providing sufficient data for deep learning network. For example, the Real-world Affective Face Database (RAF-DB) including 29,672 images that annotated into 6 basic expressions[2] plus neutral and 12 compound expressions[1]. Same as web database, Emotion Net includes 1,000,000 images that are classified into 23 basic expressions or compound expressions. Another contributing factor is "increased chip processing abilities and well-designed network architecture"[1], leading facial expression recognition to deep learning method.

As mentioned before, RAF-DB contains 29,672 real-world facial images from Internet. These images are divided into two subsets: single-label subset, including seven classes of basic emotions, and two-tabs subset, adding twelve classes of compound emotions. In this study, we only used single-label subset, which contains 15,339 images that are annotated into seven types of labels: "Surprised", "Fearful", "Disgusted", "Happy", "Sad", "Angry", and "Neural". Furthermore, single-label subset is split into two groups: 12,271 training images and 3,068 testing images. It is worth mentioning that RAF-DB adopts crowd-sourcing method to annotate images. 315 annotators label images into one of seven basic emotions, and each image is annotated around 40 times. As a result, RAF-DB is relatively reliable.

2. Description of your individual work

CNN is mostly applied to images recognition. CNN is made of three kinds of layers: convolutional layer, pooling layer, and fully connected layer. As shown is figure 2, the convolutional layer contains a set of kernels to convolve through the whole picture, thus each kernel generates one feature map. One characteristic of CNN is sharing weight (kernel), different from MLP. The convolutional layer is followed by pooling layer, which reduces the size of feature map and computational cost. There are two common method called average pooling and max pooling. The last layer of CNN is fully connected layer. Due to the input of Softmax must be vector, matrix needs to be converted to vector by fully connected layer.

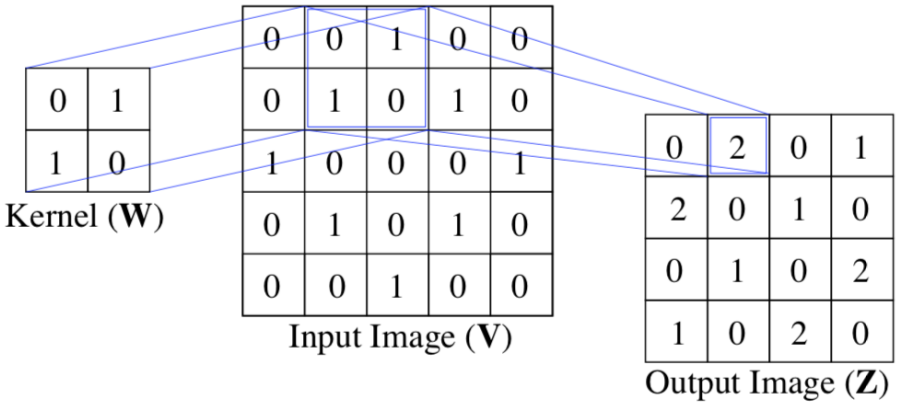


Figure 2. Kernel convolves through image

3. Description of my portion of work

For convolution neural network, we defined our models in V2_models.py file then import them in V2_main.py file:

```
model = V2_models.first_model()
```

We used keras and tensorboard for real time model evaluation.

For the first model, we added two convolution layers with kernel size of 3 and the activation function ReLu. The filter sizes are 64 and 32. A fully connected layer is followed with the softmax classifier. We set the batch size of 32 and epoch number of 200. The validation accuracy of the first model is 65%. So, a maxpooling layer is added to the second model to reduce spacial dimensions. However, with the same batch size and epoch number, the validation accuracy only increased from 65% to 66%. Then we modified the filter size of two convolution layers to both 128. The deeper we go in the network, the smaller the spatial dimensions of our volume, and the more filters we learn. With larger filter size, the training time increased for each epoch. So, we set the epoch number to 100. But the validation accuracy converged around 66% within 100 epochs. Then we considered to add dropout layers and one more fully connected layers to ensure we do not reduce our output size too quickly. The new fully connected layer is specified with a revised linear unit activation.

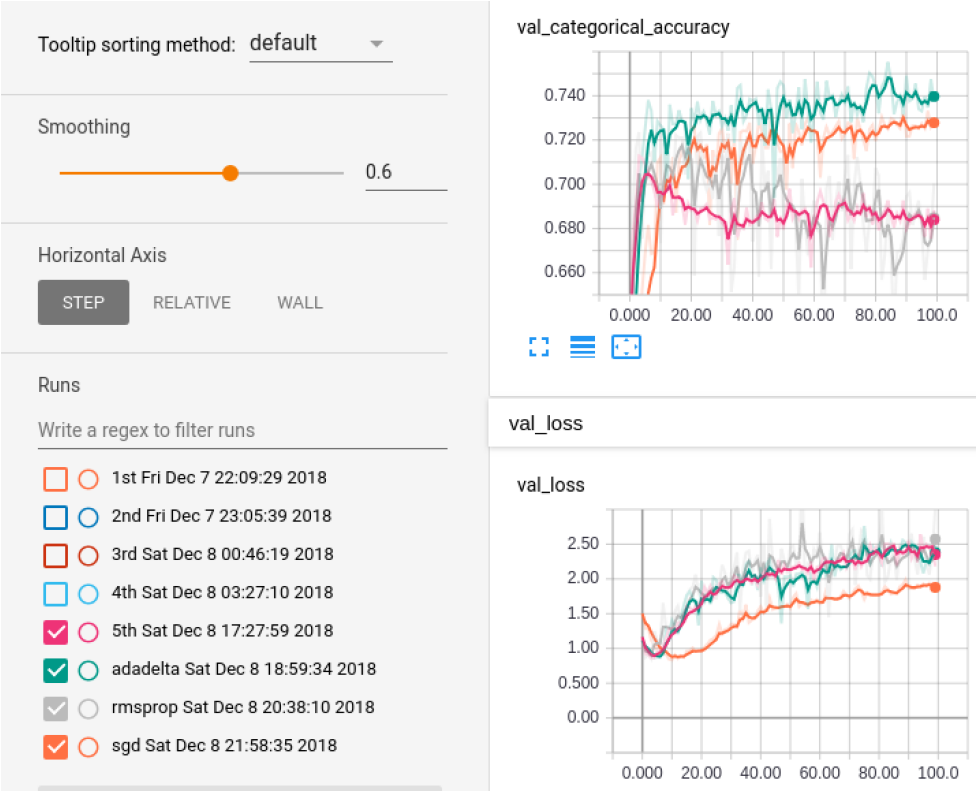
```
layers.Dropout(0.1),
layers.Flatten(),
layers.Dense(32, activation='relu'),
layers.Dropout(0.1),
```

The validation accuracy increased to 66.62% which is not that good as we expected. So we considered there might be something wrong with optimizer. We tried with four common optimizers Adam, Adadelta, RMSprop and SGD with learning rate decay and concluded with the top validation accuracy of 74%.

4. Results

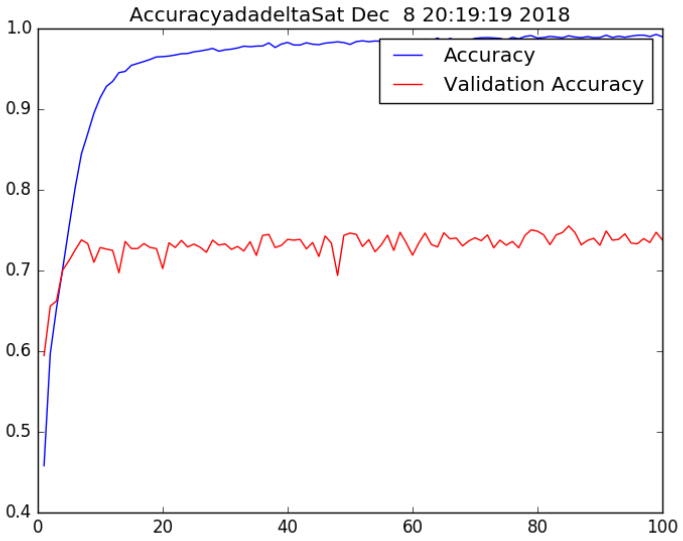
During the first four models, we made some adjustments of model structures including adding maxpooling layer(2nd), modified filter size(3rd) and adding fully connected and dropout layers(4th).

the figure above, we could see the 4th model performance better than the other three with the validation accuracy around 66%. Then we take a look at the other four models which make adjustments of keras optimizer.



With a learning rate decay of 1.0, the Adadelta optimizer performanced better than others in validation accuracy. While considering both validation loss and accuracy, we could see the performance of SGD optimizer is impressive as well from the figure above.

5. Summary and conclusions



From the result above, we could see that the best way to approach the RAF-DB single-label subset using convolution neural network is the Adadelta model. It reaches the highest accuracy among CNN models with the total one of 99% and the validation one of 74%.

6. Percentage calculation

Code from internet	modified	Own code	percentage
36	13	119	14.8%

7. References

1) Deep Facial Expression Recognition: A Survey, access at https://arxiv.org/abs/1804.08348 on Oct 22, 2018.
2) Universals and Cultural Differences in the Judgments of Facial Expressions of Emotion, access at https://www.paulekman.com/wp-content/uploads/2013/07/Universals-And-Cultural-Differences-In-The-Judgment-Of-Facia.pdf .